UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/903,947	07/12/2001	Matthew Edward Aubertine	AUS9-2000-0328US1	1975

7590 03/10/2005

SAWYER LAW GROUP
P.O. Box 51418
Palo Alto, CA 94303

EXAMINER	
VU, TUAN A	

ART UNIT	PAPER NUMBER
2124	

DATE MAILED: 03/10/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/903,947	**Applicant(s)** AUBERTINE, MATTHEW EDWARD	
	Examiner Tuan A Vu	**Art Unit** 2124	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 October 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☐ Claim(s) _____ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-15* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application (PTO-152)

6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to the Applicant's response filed 10/19/2004.

As indicated in Applicant's response, no claims have been amended. Claims 1-15 are pending in the office action.

Double Patenting

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 5, 10, 15 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 6, and 18 of copending Application No. 09/903937 (hereinafter '937). Although the conflicting claims are not identical, they are not patentably distinct from each other because of the following observations.

Following is some examples of conflicting claims:

As per instant claim 5, copending '937 claim 6 also recites a method including a plurality of directories, such method for compiling a file in the directories, and providing a master array of dependencies of the directories (e.g. *merging the dependency array with the master array of changes*); providing a code change to provide an updated program (e.g.

providing a directory update mechanism); providing associated dependency changes to the master array; and steps for

providing (dependency changes),

merging (with master array) ;

obtaining (dependency change);

determining (change is in a directory)

updating (directory in the master array)

adding (dependency change)

determining (another dependency change)

repeating (steps …until all dependency..).

But ‘937 claim 6 does not recite compiling the updated program utilizing the updated master array wherein the directories file are compiled in an ordered manner based upon the dependencies of the pluralities of directories.

However, ‘937 claim 6 recites an update mechanism for assigning the directories wherein the associated dependencies are provided and subsequent update changes merged into the master array and assigning a directory to a next available processor in an ordered manner to allow the processor to compile one file in the directory; hence suggests a program operable with a plurality of directories (i.e. *update mechanism program*) being adjusted to enable file in a directory to be compiled in an ordered manner according to the dependencies changes provided to the program. Hence, it would have been obvious for one skill in the art at the time the invention was made to provide the update mechanism as a compilable software program, and compiling of such updated program (upon dependencies changes being made thereto) so to utilize the updated dependencies (in the master array) for enabling an ordered manner of file compilation based thereupon.

As per instant claim 10, '937 claim 18 recites the same limitations and obvious variations of claim 10 (e.g. compiling an updated program); all of these features having been addressed above.

As per instant claim 15 (a computer-readable medium version of claim 10), '937 claim 18 recites the same limitations and obvious variations of claim 10 as addressed above.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claims 1, 3-4, 6, 8-9, 11, and 13-14 are rejected under 35 U.S.C. 102(b) as being anticipated by Kionka, USPN: 5,361,357 (hereinafter Kionka).

As per claim 1, Kionka discloses a method for minimizing the cycle time when compiling a program in a computer system (e.g. *optimizing, efficient compilation* - col. 1, lines 7-25), the program including a plurality of directories (e.g. Fig. 2a) and each of the directories including a code file; the method comprising:

providing a master array of directories of the program (e.g. *abstracted tree register 48* – col. 6, lines 21-68; Fig. 2C), wherein the master array lists the dependencies of the directories, providing a code change to the program to provide an updated program (e.g. *super Makefile; modified smf* – col. 7-8; *groups of Makefiles out-of-date, doOneMakefile* – col. 11-12 – Note: compounding individual Make into an updated one Make file is updating with code change);

providing associated dependency changes to the master array to provide an updated master array (e.g. *are to be updated* - col. 6, lines 60-68); and

But Kionka does not explicitly disclose compiling the updated program utilizing the updated master array wherein the code files of the directories are compiled in an ordered manner based upon the dependencies of the plurality of directories. However, Kionka discloses a equivalent of a master Makefile integrating all the individual directory Makefiles (e.g. col. 6, lines 31-41; Appendix – col. 7-26); hence has implicitly disclosed compiling of an updated and master Makefile using the order of dependencies as adjusted from integrating the plurality of directories set forth in the individual Makefiles. Therefore, the limitation is disclosed.

As per claim 3, Kionka discloses that the associated dependencies changes are provided via a directory update mechanism (e.g. *...are to be updated* -- col. 6, lines 42-68; col. 7-8; col. 11-12).

As per claim 4, Kionka discloses the steps of:

providing an array of dependency changes (e.g. *Directory Description register 50* - Fig. 2a-c; registers *40, 46, 56* – Fig. 1; col. 5, lines 60-67); and

merging the dependency changes array with a master array of changes (e.g. *Abstract Tree Register 48, Output file Register 54* – Fig. 1).

As per claim 6, Kionka disclose a system for minimizing the cycle time when compiling a program in a computer system, the program including a plurality of directories and each of the directories including a code file, the method comprises the steps of

providing (a master array),

providing (an updated program);

providing (dependencies changes ... updated master array); and

compiling (the updated program ...an ordered manner); all of which steps having been addressed in claim 1.

As per claims 8-9, these claims correspond to claims 3-4; hence are rejected using the same rejection as set forth therein, respectively.

As per claim 11, this is the computer-readable medium version of method claim 1, including the same step limitations; hence is rejected using the corresponding rejection as set forth therein.

As per claims 13-14, these claims correspond to claims 3-4; hence are rejected using the same rejection as set forth therein, respectively.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 2, 7, 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kionka, USPN: 5,361,357 (hereinafter Kionka), as applied to claims 1, 6, 11; in view of Hanna et al.,

USPN: 5,748,961(hereinafter Hanna), and further in view of Chase, Jr. et al., USPN: 4,951,192 (hereinafter Chase).

As per claim 2, Kionka does not explicitly teach a scheduler being utilized to compile the updated program, wherein the scheduler receives the dependency changes and a list of processors from a processor array. Kionka, however, mentions about complexity of a system with a larger amount of dependencies and LAN connectivity with more than one processors to distribute storage burden (e.g. Fig. 1). Hanna, in a method to optimize compiling and expedite the object code building in a large system, using a Makefile (ch. 9.2 - col. 41) in conjunction with build models or a tree structure of directory models (e.g. Fig. 1a-b) analogous to a master array of directories, teaches a schedule process as to how to distribute the code building process (ch. 10.6 – col. 45). The distribution of resources and tasks in a multi-computing network or complex interconnected system using a resource scheduler as suggested by Hanna is further enhanced via the teachings by Chase to provide parallel compilation. Chase, in a system to configure a large software system analogous to Kionka or Hanna, discloses a scheduler to choose and assign available or most suitable processors from a displayed list to compile the buildable components (e.g. col. 1, line 67 to col. 2, line 25). It would have been obvious for one of ordinary skill in the art at the time the invention was made to enhance the complex system connecting a plural computer for handling resources as suggested by Kionka and Hanna so that a scheduling process as taught by Hanna can further be used in compiling task assignment via selection from a plurality of processors being listed as taught by Chase because that way more resources can be distributed and tasks can be imparted separately alleviating thereby the condition of dependency between subsystem task that might otherwise be a limiting factor to expediting the building process and product delivery of a large software system (see Chase: *impairs productivity, independent components ... parallel* - col. 1-2).

As per claims 7 and 12, these are claims corresponding to claim 2, and are rejected using the same rationale as set forth therein.

8. Claims 5, 10, and 15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kionka, USPN: 5,361,357, as applied to claims 4, 9, 14; in view of Hanna et al., USPN: 5,748,961.

As per claim 5, Kionka discloses the steps of:

obtaining a dependency change from the dependency changes array (e.g. step *29* -Fig. 2a-c; *Input: %rels dependencies for each target* -- col. 19-20; 21-22 – Note: code change in Makefile is equivalent to obtaining a dependency change, a set of dependent elements being sequenced into a list, when processing the directories);

determining whether the dependency change is in a directory in the master array (e.g. *exists, die (Not implemented), $fake_top_name not needed* – col. 21-22 – Note: making adjustment in the master Makefile is equivalent to implementing changes being reflected in the directory element comprising the master array);

(i) updating the directory in the master array of the dependency change in a directory of the master array (e.g. col. 6, lines 60-67);

(iii) determining if there is another dependency change in the dependency changes array after step (i) (see Fig. 2a-c; Appendix code with *foreach*) ; and

repeating steps (i) and (iii) until all dependency changes have been obtained from the dependency change array.

But Kionka does not explicitly disclose the step of (ii) adding dependency change to the master array in a new directory if the dependency change is not in a directory of the master array; nor does Kionka disclose repeating of steps (i), (ii) and (iii).

However, Kionka discloses that the master array includes each directory files relationship as those directories are sequentially inputted for processing (e.g. Fig. 2a-c and related text) and teaches a the list of added dependent elements pertinent to the file directory structures being inputted to form the final master structure (e.g. *@new_list = shift (); push (@new_list, $top_target* – col. 21-22; Fig. 2a, step *31*); hence has suggested the idea of creating a list of elements coming from a specific directory and of adding a list dependency changes to the master array if the change in the array/list of dependency change has not yet been included in the master array (Note: list of dependency element, e.g. *$rels Input: %rels dependencies for each target* – col. 21-22, is equivalent to array being hierarchized according to priority and rules of a Makefile). Thus, the concept of adding a array of dependency elements or directory-related elements (re Fig. 2a, step 31) into a grouping or any hierarchy of elements being stored and arrayed is suggested, i.e. a hierarchy of dependency being grouped and arrayed by its directory root. Further, Hanna, in the system as mentioned in claim 1, supports grouping of elements into the models in terms of directories (e.g. Fig. 1b).

Based on the teachings by Hanna and on the suggested concepts and master array upgrade teachings by Kionka from above, it would have been obvious for one skill in the art at the time the invention was made to provide the creation of a new change directory within the master array as suggested by Hanna or Kionka so to include successive dependency changes or array of dependency elements into such form of directory grouping as suggested above and provide the repeating of steps from (i) to (iii) because generating a set of dependent elements being added to a directory, a separate grouping structure or a flattened hierarchy of elements would provide a better view to the abstract tree, i.e. the global master array, since this master array is purposed for enabling a facilitated perception of a more complex system wherein dependency of directories being integrated in the final build is subjected to continual and dynamic updates or adjustments.

As per claims 10 and 15, these are claims corresponding to claim 5, and are rejected using the same rationale as set forth therein.

Response to Arguments

9. Applicant's arguments filed 10/19/2004 have been fully considered but they are not persuasive. Following are Examiner's observation in regard thereto.

Rejection 35 U.S.C. § 102:

(A) Applicant has submitted that Kionka's updates to directories as cited in the Office Action are updates to files and files dependencies are not the same as directories dependencies (Appl. Rmrks, 2nd para, pg. 11). The limitation recited as 'dependencies of directories' is not elaborated sufficiently in the claim to enforce that interdependent files of a directory are precluded from being what is construed as 'directories dependencies'. As construed by broad reasonable interpretation, a file is inherent to a directory path and dependency of a directory path files; and updates of such file along with the role of an abstracted structure responsible for reflecting the changes to those files (as cited in the rejection) read on 'providing associated dependency changes ... updated master array'. There are not sufficient details in the claim to distinguish a Master Array from what is recited in Kionka's teaching of a update tree, nor is there enough clarification in the claim that dictates that such array listing of dependencies of directories would be necessary different from the abstracted tree by Kionska because dependency of directories is not construed as different from dependency of directory files inherently listed in a path being one integral part of a directory. Hence, the arguments either read the specifications into the claims or amount to alleged remarks that fail to convince why Kionska's teaching would not fulfill what is explicitly recited.

Rejection 35 U.S.C. § 103:

(B) Applicant has submitted that the rejections (Appl. Rmrks, 2nd para, pg. 14) are disagreed upon based on the arguments set forth for the independent claims 1, 6, and 11. These rejections will stand as set forth on the grounds of the counter-arguments in section A above.

Conclusion

10. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kakali Chaki can be reached on (571)272-3719.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273-3735 (for non-official correspondence – please consult Examiner before using) or 703-872-9306 (for official correspondence) or redirected to customer service at 571-272-3609.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

VAT
February 25, 2005

KAKALI CHAKI
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100